1К

УДК 004.932+621.397

А.Г. Кузнецов
Ижевский государственный технический университет, Российская Федерация
po@istu.ru

Контурный метод как дополнительный этап сжатия цифровых изображений в телекоммуникационных системах

В данной статье рассмотрено новое применение техники интерполяции изображений при контурном кодировании и сжатии изображений. Также приведен алгоритм сжатия цифровых изображений, основывающийся на кодировании объектов произвольной формы. Эта процедура может быть использована в полном алгоритме сжатия цифровых изображений в качестве одного из дополнительных этапов, в результате которой увеличивается результирующий коэффициент кодирования.

Кодирование изображения контурным методом с помощью цифровой интерполяции

Как указывал Марр [1], контур всегда играл ключевую роль в зрительном восприятии и анализе изображений от классической теории «пересечения нулевого уровня» до более новой волновой теории (вейвлеты, **wavelets**). В кодировании изображений, например, производительность программы кодирования сильно обусловлена ее реакцией на резкие границы контуров. Данные моменты хорошо отражены в современном направленном развитии теории вейвлетов для кодирования изображений: изобретение **curvelets** и **beamlets** школой Донохо [2], [3], **bandlets** Пеннека и Моллата [4] и схема древовидного кодирования Кохена и др. [5].

Рассмотрим теперь применение интерполяции изображений для кодирования (компрессии) и сжатия изображения с потерями, основанной на информации о контурах в изображении.

Схема кодирования данным методом состоит из трех этапов:

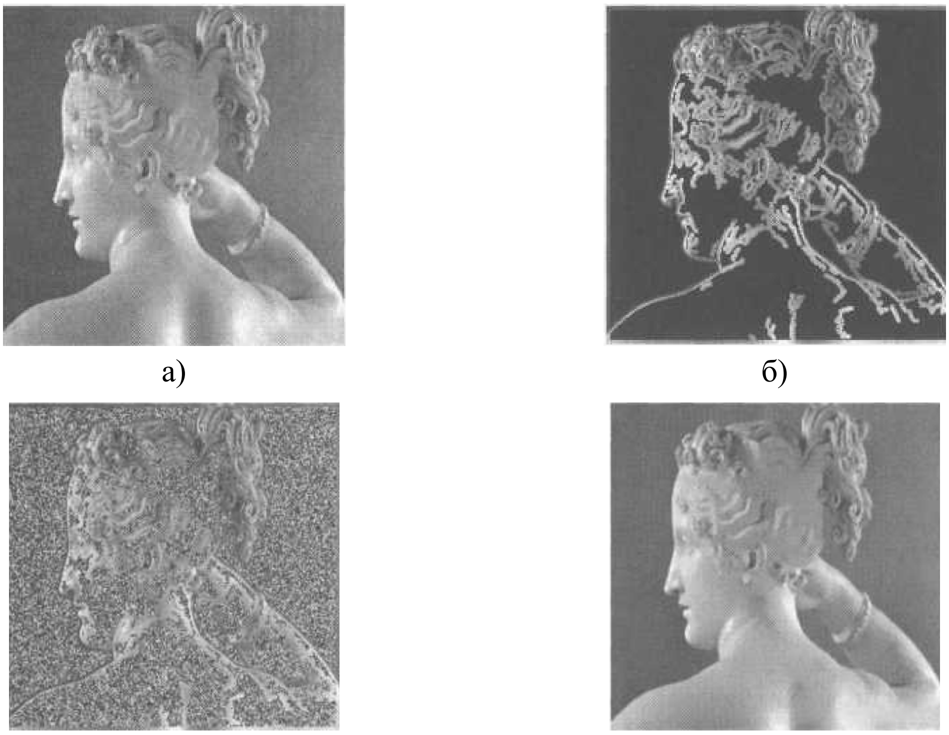
1. Обнаружение контура E. Чтобы определить набор контуров E данного изображения u^0, применяется детектор контура (например, детектор контура Канни). E – это набор пикселей или кривых оцифрованного изображения без каких-либо устойчивых геометрических закономерностей. К тому же, также необходимо ввести требование физического ограничения на набор контуров на множестве точек изображения Ω.

2. Утолщенные контуры T. Устанавливается малая константа ε, генерируется ε-окрестность T на наборе контуров. Результат и называется утолщенными контурами. Фактически T может представлять из себя 1 – 2 пиксельное утолщение E (рис. 1).

3. Кодирование. Этот этап и является собственно кодированием изображения. Кодируются адреса пикселей утолщенных контуров T и с высокой скоростью кодируется изображение в градациях серого цвета на утолщенном контуре $u^0\big|_T$ ($u^0\big|_T$ – это часть изображения, определяемая набором пикселей, входящих в T).

Этот метод кодирования создает большие зоны «пустых областей» – это такие области, где данные изображения оказываются как бы «стерты». Таким образом, за счет такой потери достигается высокая степень сжатия изображения. При отсутствии сложных

текстур и мелких деталей на изображении набор контуров состоит из одномерных кусочно-гладких кривых. Таким образом, если ε стремится к нулю, то толщина утолщенных контуров T также стремится к нулю, что теоретически приводит к бесконечной степени сжатия изображения. Естественно, такая высокая степень сжатия сильно усложняет процесс декодирования изображения (декомпрессии). Для декодирования используется метод цифровой интерполяции (ЦИ), позволяющий «закрасить» незакодированные недостающие участки.



а) б)

в) г)

Рисунок 1 – Процесс кодирования и декодирования изображения контурным методом с помощью интерполяции изображения:
а) исходное изображение u^0; б) утолщенные контуры после детектора Канни Т;
в) закодированное изображение $u^0\big|_T$; г) декодированное изображение

Для декодирования изображения применяется цифровая интерполяционная модель для утолщенных контуров T и областей изображения в градациях серого цвета, соответствующих утолщенным контурам $u^0\big|_T$:

$$\min_u \left[\sum_{\alpha\in\Omega} |\nabla_\alpha u| + \sum_{\alpha\in\Omega} \frac{\lambda_T(\alpha)}{2}\left(u_\alpha - u_\alpha^0\right)^2 \right], \quad (1)$$

где множитель Лагранжа: $\lambda_T(\alpha) = \begin{cases} \lambda, & \alpha \in T; \\ 0, & \alpha \in \Omega \setminus T. \end{cases}$, $|\nabla_\alpha u| = \sqrt{\sum_{\beta\in N_\alpha} \left(u_\beta - u_\alpha\right)^2}$ – локальные вариации, где N_α – окрестность пикселя α.

В отличие от *JPEG* или *JPEG*2000, здесь декодирование изображения было реализовано с помощью вариационной реконструкции (воссоздания) вместо обратного преобразования, типа дискретного косинусного преобразования (в *JPEG*) или быстрого вейвлет-преобразования (в *JPEG*2000).

Использование метода ЦИ в данном случае имеет еще дополнительное значение. Так как на этапе кодирования не вносится ограничение на условие регулярности набора контуров, то, как правило, E – это достаточно беспорядочный набор контуров без каких-либо устойчивых геометрических закономерностей. Таким образом, ЦИ в процессе декодирования изображения может выправить (выпрямить) волнистые контуры и, тем самым, улучшить качество визуального восприятия этих контуров.

На рис. 1 приведен процесс кодирования и декодирования изображения контурным методом с помощью интерполяции изображения.

В работе предложено новое применение метода цифровой интерполяции изображений и контурного кодирования для сжатия изображений. При высокой степени сжатия изображения происходит потеря некоторых деталей исходного изображения. Но, что примечательно, данный метод достаточно точно захватывает наиболее важную визуальную информацию изображения.

Объектное кодирование как дополнительный этап сжатия цифровых изображений

Контурный анализ изображений.

Процесс выделения контура происходит в два этапа.

1. Изображение обрабатывается (*LoG*-изображение) с помощью оператора Лапласа – Гаусса (*LoG*), который определяется следующим образом:

$$LoG(x,y)=\left(\frac{r^2-\sigma^2}{\sigma^4}\right)\exp\left(-\frac{r^2}{2\sigma^2}\right), \tag{2}$$

где $r^2=x^2+y^2$, σ – стандартное отклонение.

Чтобы упростить применение уравнения (2) к изображению, можно его предварительно упразднить различными методами.

2. Обнаруживаются фронты в точках нулевого пересечения (например, вдоль горизонтального и вертикального направлений) [6].

На этом этапе изменения оператора *LoG*-изображения вдоль направлений x и y, обозначаемых как S_x и S_y, используются для вычисления мощности фронта в точках нулевого пересечения. Мощность фронта в точке (x, y) определяется следующим образом:

$$S(x,y)=\begin{cases}\sqrt{S_x^2+S_y^2},\\ 0.\end{cases} \tag{3}$$

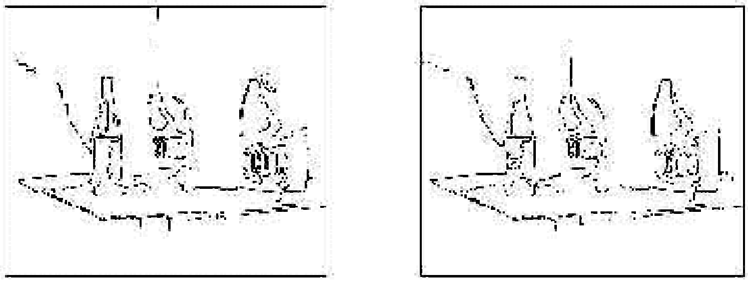
Точки контура выбираются, используя методику порогового значения гистерезиса, которая определяется следующим образом: мощность фронта в потенциальной точке вдоль контура должна быть больше, чем T_1, и, по крайней мере, одна точка контура должна иметь мощность фронта больше, чем T_2, где T_1 и T_2 – пороги, которые устанавливаются предварительно, причем $T_1 < T_2$. Обычно T_1 устанавливается на достаточно низком уровне, чтобы сохранить целостность контура возле области границы, а T_2 наоборот – выбирается большим для избежания ложных фронтов. Поиск контура повторяется всякий раз, когда одна точка со значением выше T_2 найдена. Поиск проводится в обоих направлениях контура. Соседние пикселы со значением выше T_1 также принимаются как точки контура. Поиск завершается, когда ни один найденный соседний пиксел не удовлетворяет вышеоговоренному условию. Далее все значения мощности фронтов вдоль обнаруженного контура обнуляются, чтобы не быть рассмотренными вновь в будущем. Подобная процедура поиска контура продолжается, пока всё множество значений мощности фронтов не будет проверено.

Таким образом, в результате точки обнаруженных фронтов, чьи мощности были обнулены, в дальнейшем в процедуре кодирования цифрового изображения не используются, тем самым, удаляя избыточную информацию и повышая результирующий коэффициент сжатия.

Пример изображения, к которому применен рассмотренный выше алгоритм, и найденные в результате вышеописанных операций контуры проиллюстрированы на рис. 2.



а)



б)

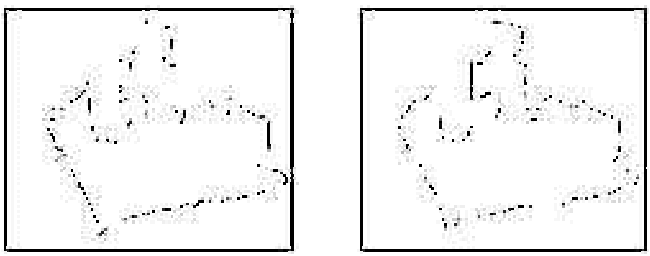
Рисунок 2 – а) исходные изображения; б) найденные контуры

Контурные определения.

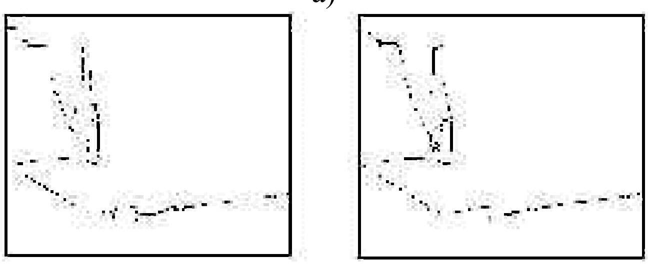
Оконтуренные объекты определяются в терминах:

1) те, которые замкнуты, как показано на рис. 3а;

2) те, которые открыты и заканчиваются на границах изображения, как показано на рис. 3б.



а)



б)

Рисунок 3 – а) замкнутые контуры; б) открытые контуры

Определение замкнутого контура.

Для каждого замкнутого контура вычисляются три атрибута формы:
– количество пикселей, составляющих периметр контура, n;
– y задает размещение центроида (x_c, y_c);
– первый инвариантный момент h (инвариантный к преобразованию, вращению и изменению масштаба). Следует обратить внимание, что y_c рассматривается как хороший признак определения, поскольку мы предполагаем, что вертикальное несоответствие между точками в структуре изображения незначительно. Это потому, что параллельная геометрия оси используется в получении всех проверочных цифровых изображений. Если x_i и y_i представляют координаты x и y точек вдоль контура, то h может определяться следующим выражением:

$$h = \frac{1}{n^2}\sum_{i=1}^{n}\left[\left(x_i - x_c\right)^2 + \left(y_i - y_c\right)^2\right], \qquad (4)$$

где $x_c = \frac{1}{n}\sum_{i=1}^{n} x_i$ и $y_c = \sum_{i=1}^{n} y_i$. (5)

Каждый замкнутый контур изображения сравнивается с рассматриваемым контуром, который принимается в качестве начального кандидата для обнуления, если разница между каждым атрибутом формы оказывается ниже заранее установленного порога (например, 20 % для h и 10 % для y_0 и n). В конце данной процедуры среди замкнутых контуров может быть несколько кандидатов для сравнения. И в дальнейшем необходимым шагом является поиск наиболее подходящего контура среди них. Это выполняется следующим образом.

Во-первых, контуры преобразуются в их цепной код $\{a_i \in \{0,1,2,...,7\}\}$. Цепной код – это наиболее сжатый способ представления контура по сравнению с простым набором координат точек вдоль контура. Он описывает контур как ряд однопиксельных векторных переходов в различных направлениях (8 различных направлений для 8-связных контуров) из начальной точки, определенной однозначно. В экспериментах в качестве стартовой точки выбирался самый крайний левый верхний пиксел в данном контуре. Однако окончательный результат мог бы быть независимым от этой начальной точки. Чтобы избежать так называемой проблемы замыкания по периметру в стандартный цепной код представлений, вначале необходимо трансформировать данный стандартный цепной код $\{a_1 a_2 ... a_n\}$ длиной n в модифицированный код $\{b_1 b_2 ... b_n\}$ рекурсивной операцией замены, которая опередяется следующим образом:

$$b_1 = a_1,$$
$$b_i = q_i,$$

где q_i – целое число такое, что: $\left(q_i - a_i\right)\text{mod}\,8 = 0$ и $|\,q_i - b_{i-1}\,|$ – минимально; $I = 1, 2, 3, \dots, n$.

Положим, что $\{l_i\}$ и $\{r_i\}$ есть представления цепного кода двух контуров L и R (модифицированных, как описано выше), которые находятся в процессе сравнения, соответственно, N_L и N_R – их длины. Мы можем определить меру корреляции MC_{kl} между двумя n-точечными участками, одним началом в индексе k контура L и другим началом в индексе l контура R. Нестандартное определение MC_{kl} дано ниже:

$$MC_{kl} = \frac{1}{n}\sum_{j=0}^{n-1}\cos\frac{\pi}{4}\left(l'_{k+j} - r'_{l+j}\right), \qquad (6)$$

где $l'_{k+i} = l_{(k+i)\,\text{mod}\,N_L} - \frac{1}{n}\sum_{j=0}^{n-1} l_{(k+j)\,\text{mod}\,N_L}$, $r'_{l+i} = r_{(l+i)\,\text{mod}\,N_R} - \frac{1}{n}\sum_{j=0}^{n-1} r_{(l+j)\,\text{mod}\,N_R}$, $0 \le i < n$.

Чтобы определить местоположение наиболее подходящего среди двух контуров, n-точечного сегмента контура L, начала в индексе k (например, $k = 0$), обходим контур R. Подобная функция:

$$F_{LR} = \max\{MC_{kl}\},\ l \in M\ ,$$

где M определяет диапазон поиска, используется затем как раз для определения наиболее подходящего местоположения.

Контур R выбирается в случае, если выполняются следующие два условия:

1) $F_{LR} \geq F_{L'R}$, где L' представляет все контуры со схожими формами (соответствия первоначальных претендентов) в контур R;

2) $F_{LR} > T$, где T – заранее заданный порог, который исключает соответствия с плохой корреляцией, иначе выбирается контур L.

В случае, когда среди множества отобраны контуры, контур с максимумом F_{LR} выбирается как подходящий.

Определение открытого контура.

Для соответствия открытых контуров ярко выраженные участки (выпуклые, выступающие, выделяющиеся) вдоль контуров используются как соответствующие примитивы. Эти участки, такие, как углы, могут быть определены из цепного кода изображения. Если более конкретно, то для контура длиной n с цепным кодом $\{l_i\}$ величина кривизны в i-й точке определяется как [7], [8]:

$$C_i = \max_{1 \leq j \leq 3\sigma}\{\max\{|\, l_{i-j} - l_{i+j}\,|, |\, l_{i-j} - l_{i+j-1}\,|\}\}\,, \tag{7}$$

где σ – стандартное отклонение оператора LoG, используемое для выделения контура.; i-я точка вдоль контура выбирается как выделяющаяся точка, если выполняются оба следующих условия:

$$1)\ c_i \geq T_S\ \text{ и } 2)\ c_i \geq c_k\ \text{ для всех } k \in [i-p, i+p],$$

где p – константа, которая определяет минимальное расстояние между выделяющимися точками;

T_S – порог, определяющий минимальную приемлемую кривизну.

Участки контура, окружающие значащие точки, затем используются как шаблоны 1-D в нахождении подходящих соответствий в другом изображении.

Запирание контура.

Запирание контура – процесс, разработанный для того, чтобы самые эффективные способы подбора объектов внутри блоков могли быть применены в различных методах сжатия изображений. Основной подход данного процесса включает размещение объекта в наименее возможный прямоугольник, добавляя ссылку (связь) на объект, и выполнение компенсации предсказываемых ошибок.

Кодирование объекта.

Объектный способ кодирования цифровых изображений определяет три типа объектов на изображениях: объекты, которые находятся внутри замкнутых контуров; объекты, которые имеют открытые контуры и оба конца открытого контура заканчиваются на границе изображения; и остальные объекты, которые относятся к фону. Хотя и есть отличия среди этих трех типов, и кодирование их требует индивидуального подхода, предложенный алгоритм кодирования может быть описан следующим образом:

1) дополнение объекта левыми ограничивающими прямоугольниками;

2) основанное на неравенстве предсказание для обоих блоков произвольной формы и внутренних блоков;

3) кодирование фона.

Дополнительное примечание к полной схеме кодирования.

1. Дополнение объекта левыми ограничивающими прямоугольниками представляет локальный градиент внутри объекта.

2. Значения несоответствия и предсказываемые ошибки для каждого блока произвольной формы передаются объекту на основе объекта. Таким образом, ошибка блоков и значение несоответствия могут быть правильно определены и использованы для восстановления объектов в конце декодирования.

3. В процессе предсказания форма объектов в изображении используется для того, чтобы производить ошибки для объектов и, таким образом, никакие биты не используются для кодирования формы объекта. Всё же правильная форма может быть восстановлена в конце декодирования, т.к. потерянная информация будет заменена фоном соответствующего кодируемого блока.

4. Внутренние области кодируются относительно (адаптивно) их местной структуре.

Литература

1. Marr D. – Vision. – Freeman: San Francisco, 1980.
2. Candes E.J., Donoho D.L. Curvelets and reconstruction of image from noisy radon data // Wavelet Applications in Signal and Image Processing VIII / A. Aldroubi, A.F. Laine, M.A. Unser, eds. – Proc. SPIE 4119, 2000.
3. Donoho D.L., Huo X. Beamlets and multiscale image analysis // Multiscale and Multiresolution Methods. – Vol. 20 of Lect. Notes Comput. Sci. Eng. – Springer-Verlag, Berlin, 2002. – P. 149-196.
4. Pennec E.L., Mallat S. Image compression with geometrical wavelets // Proc. Of IEEE ICIP. – 2000. – Vol. 1. – P. 661-664.
5. Cohen A., Dahmen W., Daubechies I., DeVore R. Harmonic analisys of the Space BV. – Rev. Mat. Iberoamericana. – 2003. – № 19. – P. 235-263.
6. Chan Tony F., Shen J. Image Processing and analysis: variational, PDE, wavelet, and stochastic methods. – 2005. – 402 p.
7. Pennec E.L., Mallat S. Image compression with geometrical wavelets // Proc. Of IEEE ICIP. – 2000 – Vol. 1. – P. 661-664.
8. Wallach D.S., Kunapalli S., Cohen M.F. Accelerated MPEG compression of Dynamic poligonal scenes // ACM. – June 1994.
9. Mallat S. A theory for multiresolutional signal decomposition: the wavelet representation // IEEE Trans. Pattern Analysis and Machine Intelligence. – 1989. – № 7. – P. 674-693.

А.Г. Кузнецов

Контурний метод як додатковий етап стиснення цифрових зображень у телекомунікаційних системах

У даній статті розглянуто нове використання техніки інтерполяції зображень при контурному кодуванні та стисненні зображень. Також наведений алгоритм стиснення цифрових зображень, який ґрунтується на кодуванні об’єктів будь-якої форми. Ця процедура може бути використана у повному алгоритмі стиснення цифрових зображень у якості одного з додаткових етапів, в результаті якої збільшується результуючий коефіцієнт кодування.

Статья поступила в редакцию 02.07.2008.